# Diseño e implementación de una red de Comunicaciones

Juan Enrique Rozo Tarache
Depto. Ingeniería de Sistemas
Pontificia Universidad Javeriana
Bogotá, Colombia
juan.rozot@javeriana.edu.co

Julián Camilo Ramos Granada
Depto. Ingeniería de Sistemas
Pontificia Universidad Javeriana
Bogotá, Colombia
julianc.ramos@javeriana.edu.co

Daniel Felipe Castro Moreno
Depto. Ingeniería de Sistemas
Pontificia Universidad Javeriana
Bogotá, Colombia
castro-df@javeriana.edu.co

María Paula Rodríguez Ruiz
Depto. Ingeniería de Sistemas
Pontificia Universidad Javeriana
Bogotá, Colombia
rodriguezr.mp@javeriana.edu.co

***Abstract*— In the business landscape, network connectivity and security are paramount for organizational success. Company XYZ, specializing in software development and IT consulting, is overhauling its network infrastructure. Thus, the Systems Department is tasked with devising a data network design capable of meeting current demands while also accommodating future expansions and technological advancements seamlessly. This design addresses the growing need for mobility, flexibility, and enhanced security, along with ensuring web services availability and network optimization. A prototype utilizing the GNS3 platform is proposed to provide internet connectivity to hosts across two buildings housing management, development, system, and research departments, with only the latter being compatible with both IPv4 and IPv6. Additionally, the implementation of NAT for custom IP routing, VLANs for inter-link I/O, and redundancy and network security segmentation techniques are discussed for optimal performance. Various server categories (WEB, DHCP, LDAP, and DNS) are highlighted, along with their usage and technical specifications, tailored to meet the specific requirements of company XYZ.**

**Keywords— GNS3, IPv4, IPv6, NAT, network design, network segmentation, redundancy, security, server implementation, VLAN.**

## I. INTRODUCCIÓN

En el entorno empresarial actual, la conectividad y seguridad de la red son esenciales para que cualquier organización tenga éxito. La empresa XYZ, que se especializa en el desarrollo de software y consultoría en servicios informáticos, está actualizando su infraestructura de red y se le ha encargado al Departamento de Sistemas proponer un diseño de red de datos que no solo pueda respaldar las operaciones actuales, sino también preparar el terreno para futuras expansiones y demandas tecnológicas; es decir, permitir una fácil actualización.

Dentro de la organización, este diseño debe adaptarse a la creciente necesidad de movilidad y flexibilidad al mismo tiempo que incorpora elementos como seguridad, disponibilidad de servicios y rendimiento optimizado de la red. Por lo tanto, se presenta la implementación de un prototipo utilizando la plataforma GNS3, donde se da conectividad a internet a todos los hosts, dentro de los 2 edificios de 2 pisos cada uno, repartidos entre los departamentos de Gerencia, Desarrollo, Sistemas e Investigación, siendo este último el único con compatibilidad tanto para IPv4 como IPv6.

En adición, se presenta la documentación e implementación de NAT para un esquema de direccionamiento de IP propio, VLAN para la comunicación entre hosts y enlaces redundantes y segmentación de Redes para garantizar una óptima seguridad. Finalmente, se exponen servidores (WEB, DHCP, LDAP y DNS, su uso y detalles técnicos para satisfacer las necesidades específicas de la empresa XYZ.

## II. DISEÑO DE RED

En un diseño de red se abarca una amplia gama de dispositivos, modelos y protocolos que desempeñan un papel fundamental en el intercambio de información entre sistemas y equipos. Como primera medida, se explicará detalladamente el funcionamiento de algunos componentes esenciales en el diseño de una red, con el fin de tener una mayor comprensión teórica de los temas que se abordan en el diseño de la topología.

### *A. Switch*

Son dispositivos de capa 2, su función principal es reenviar tramas de un puerto a otro para que la información que circula por la red llegue a su destino. A diferencia de los hubs, que simplemente repiten las tramas a todos los puertos, un switch lo hace en función de la información que tiene aprendida sobre la red. Esta información de la red se llama tabla de direcciones MAC y es gracias a ella que se sabe qué equipo está conectado a cada puerto. Cuando un switch recibe una trama, examina su encabezado en busca de la dirección MAC de destino. Luego, comienza a buscar en la tabla de direcciones MAC y en caso de que la encuentre, reenvía la información por ese puerto. Por otro lado, si no encuentra la dirección en la tabla, se reenvía la trama por todos los puertos excepto por el que la recibió, asegurándose de que la información llegue al destinatario [8].

En caso de que la dirección MAC de destino sea una dirección ffff.ffff.ffff (broadcast), el switch no consulta la tabla de direcciones MAC sino que directamente la reenvía por todos los puertos [8].

Para el proceso de la construcción de la tabla de direcciones MAC, es importante tener en cuenta que se trata de un proceso automático. Consiste en que cuando un equipo envía una trama

a un switch, el encabezado de la trama tendrá las direcciones de origen y destino, por lo que, si el switch detecta que la dirección de origen no se encuentra registrada en la tabla de direcciones MAC, asocia esa dirección al puerto por el que recibió la información [8].

### *B. Hub*

Un hub es un dispositivo de red de capa 1 en el modelo OSI que se utiliza para conectar varios dispositivos de una red. Tiene múltiples puertos en los que se puede conectar varios dispositivos de red, como computadores, impresoras, switches, entre otros [9].

Funcionan como repetidores de señales, lo que significa que reciben datos en un puerto y los transmiten a todos los demás puertos. Esta función hace que todos los dispositivos conectados al hub compartan el mismo dominio de colisión, lo que puede provocar congestión en la red y disminuir el rendimiento en entornos con muchos dispositivos. Por ello, en la actualidad se encuentran prácticamente descontinuados dadas sus limitaciones en rendimiento y eficiencia [8].

Las diferencias entre un switch un hub se pueden visualizar mejor en la siguiente imagen:

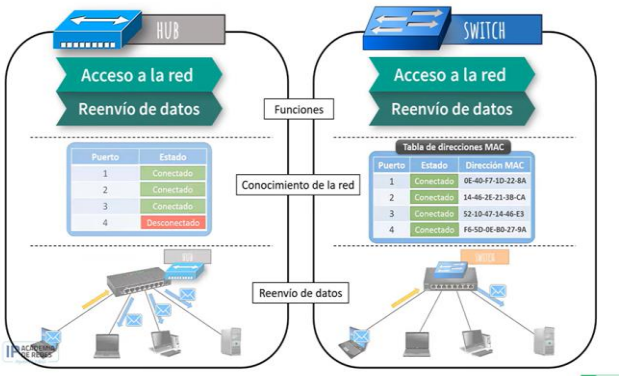


*Fig. 1. Diferencias entre switch y hub en [8].*

### *C. Router*

Son dispositivos de capa 3 capaces de leer e interpretar direcciones IP, lo que les permite redirigir paquetes de una red a otra permitiendo la comunicación entre equipos que pertenecen a direcciones IP distintas. Explicado de otra manera, funcionan como las puertas de enlace entre redes para una comunicación eficiente. Su funcionamiento se visualiza mejor en figura 2.

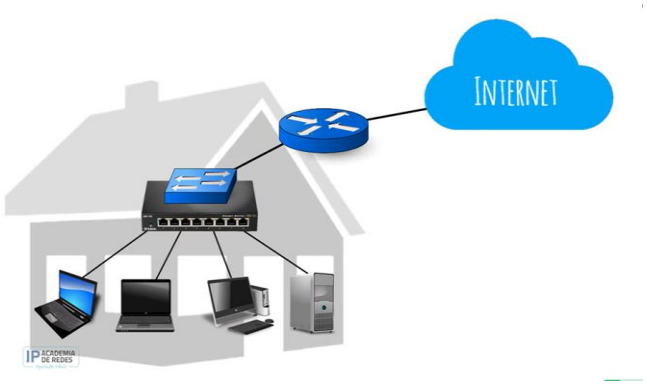


*Fig. 2. Router en una topología de red en [10].*

De igual manera, en la figura 3 se puede observar la configuración del router en GNS3.

```
IPv6#
IPv6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
IPv6(config)#interface fastEthernet 0/0
IPv6(config-if)#ip address 192.168.5.145 255.255.255.240
IPv6(config-if)#ipv6 address FD00:1::1/64
IPv6(config-if)#no shutdown
IPv6(config-if)#exit
IPv6(config)#
*Mar  1 00:02:43.427: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:02:44.427: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, change
d state to up
IPv6(config)#interface fastEthernet 0/1
IPv6(config-if)#ip address 192.168.5.161 255.255.255.240
IPv6(config-if)#ipv6 address FD00:2::1/64
IPv6(config-if)#no shutdown
IPv6(config-if)#exit
IPv6(config)#
*Mar  1 00:03:35.587: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 00:03:36.587: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, change
d state to up
IPv6(config)#ipv6 unicast-routing
IPv6(config)#
```

*Fig. 3. Configuración de las direcciones IPv4 e IPv6 del router en GNS3, elaboración propia.*

Como primer paso se accede al modo de configuración de la interfaz FastEthernet 0/1 y se habilita. Posteriormente se configura el modo de encapsulación de la interfaz FastEthernet 0/1 en acceso y se configura su dirección IP (tanto IPv4 como IPv6). Finalmente se sale de los modos de configuración y se guarda.

Por su parte en la figura 4 se hace la comprobación de las direcciones IPv4 e IPv6 del router.

```
IPv6(config)#exit
IPv6#show ip
*Mar  1 00:28:04.683: %SYS-5-CONFIG_I: Configured from console by console
IPv6#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.5.145   YES manual up                    up
FastEthernet0/1        192.168.5.161   YES manual up                    up
FastEthernet1/0        unassigned      YES unset  administratively down down
IPv6#show ipv6 interface brief
FastEthernet0/0          [up/up]
    FE80::CE01:33FF:FE74:0
    FD00:1::1
FastEthernet0/1          [up/up]
    FE80::CE01:33FF:FE74:1
    FD00:2::1
FastEthernet1/0          [administratively down/down]
IPv6#
```

*Fig. 4. Comprobación de direcciones IPv4 e IPv6 para router, elaboración propia.*

Por último, en la figura 5 se puede observar que como primer paso se accede al modo de configuración de FastEthernet 1/0 y se habilita la interfaz, para posteriormente configurar su dirección IP, salir de los modos de configuración

y guardar. La ruta de IPv4 se muestra en la figura 6 e IPv6 en la figura 7.

```
IPv6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
IPv6(config)#interface fastEthernet 1/0
IPv6(config-if)#ip address 172.16.1.2 255.255.255.0
IPv6(config-if)#ipv6 address FD00:7::2/64
IPv6(config-if)#no shutdown
IPv6(config-if)#exit
IPv6(config)#
*Mar  1 00:37:24.791: %LINK-3-UPDOWN: Interface FastEthernet1/0, changed state t
o up
*Mar  1 00:37:25.791: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et1/0, changed state to up
IPv6(config)#ip route 192.168.5.0 172.16.1.1
% Incomplete command.

IPv6(config)#ip route 192.168.5.0 255.255.255.192 172.16.1.1
IPv6(config)#ip route 192.168.5.64 255.255.255.224 172.16.1.1
IPv6(config)#ip route 192.168.5.96 255.255.255.224 172.16.1.1
IPv6(config)#ip route 192.168.5.128 255.255.255.240 172.16.1.1
IPv6(config)#ipv6 route FD00:3::/64 FD00:7::1
IPv6(config)#ipv6 route FD00:4::/64 FD00:7::1
IPv6(config)#ipv6 route FD00:5::/64 FD00:7::1
IPv6(config)#ipv6 route FD00:6::/64 FD00:7::1
IPv6(config)#exit
IPv6#d
*Mar  1 00:41:20.659: %SYS-5-CONFIG_I: Configured from console by console
IPv6#write
Building configuration...
[OK]
```

*Fig. 5. Configuración 2 del enrutamiento IPv4 e IPv6 del router en GNS3, elaboración propia.*

```
IPv6#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.1.0 is directly connected, FastEthernet1/0
     192.168.5.0/24 is variably subnetted, 6 subnets, 3 masks
S       192.168.5.96/27 [1/0] via 172.16.1.1
S       192.168.5.64/27 [1/0] via 172.16.1.1
S       192.168.5.0/26 [1/0] via 172.16.1.1
C       192.168.5.160/28 is directly connected, FastEthernet0/1
S       192.168.5.128/28 [1/0] via 172.16.1.1
C       192.168.5.144/28 is directly connected, FastEthernet0/0
```

*Fig. 6. Comprobación de enrutamiento IPv4, elaboración propia.*

```
IPv6#show ipv6 route
IPv6 Routing Table - 12 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
C   FD00:1::/64 [0/0]
     via ::, FastEthernet0/0
L   FD00:1::1/128 [0/0]
     via ::, FastEthernet0/0
C   FD00:2::/64 [0/0]
     via ::, FastEthernet0/1
L   FD00:2::1/128 [0/0]
     via ::, FastEthernet0/1
S   FD00:3::/64 [1/0]
     via FD00:7::1
S   FD00:4::/64 [1/0]
     via FD00:7::1
S   FD00:5::/64 [1/0]
     via FD00:7::1
S   FD00:6::/64 [1/0]
     via FD00:7::1
C   FD00:7::/64 [0/0]
     via ::, FastEthernet1/0
L   FD00:7::2/128 [0/0]
     via ::, FastEthernet1/0
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
     via ::, Null0
IPv6#
```

*Fig. 7.Comprobación de enrutamiento IPv6, elaboración propia.*

*D. Switch de capa 3*

Un switch de capa 3 es un dispositivo el cual es capaz de hacer las funciones de un switch de capa 2, encargado de realizar el switching a nivel de puertos, y a la vez las funciones de un router de capa 3, encargado del enrutamiento a nivel de red. La principal ventaja de un switch de capa 3 es su capacidad para realizar forwarding entre VLANs de una manera mucho más rápida que un router, ya que lo realiza vía hardware mediante el uso del chip ASICS (circuito integrado de aplicación específica). Este chip surgió en el momento en el que la industria y el mercado se dio cuenta de que los protocolos que se usaban eran siempre los mismos (ethernet e IP), dado que el mercado se estandarizó. Por lo que se desarrolló un producto que se centre en estos dos protocolos para así poder realizarlo de una forma mucho más rápida [10].

*E. Diferencias entre un switch de capa 3 y un router*

Como se mencionó anteriormente, la principal ventaja de un switch de capa 3 es su capacidad para realizar forwarding entre VLANs gracias a que lo realiza vía hardware. Por su parte, los routers tradicionalmente realizan el reenvío de paquetes utilizando Software, lo que implica que dependen de un procesador para realizar estas funciones. Si bien los routers tienen la ventaja de ofrecer una capacidad de procesamiento más amplia que los switches de capa 3, no alcanzan la misma velocidad debido al procesamiento en software. Sin embargo, algunos fabricantes están implementando routers con funcionalidades hardware para mejorar su rendimiento y competitividad [10].

En cuanto a su utilización, los routers están más orientados a escenarios de red de área amplia (WAN), donde se requiere una capacidad de enrutamiento dinámico mayor y una gestión de tablas de rutas más extensa. Por otro lado, los switches de capa 3 son más adecuados para redes de área local (LAN), donde hay una alta densidad de dispositivos conectados en una zona geográfica limitada. Aunque un switch de capa 3 puede comportarse como un router, este último sigue siendo más adecuado para entornos WAN debido a su capacidad de enrutamiento [10].

Otra diferencia a tener en cuenta es que mientras que un switch de capa 3 solo soporta Ethernet, un router ofrece una variedad de opciones de conectividad, como interfaces serie, conexiones DSL (Digital Subscriber Line), 5G, ATM (Asynchronous Transfer Mode), Frame Relay, entre otras, lo que se traduce en una mayor flexibilidad para adaptarse a diferentes tipos de redes y escenarios de implementación [10].

*F. Modelo OSI*

El modelo OSI (Open Systems Interconnection) es un modelo utilizado para diseñar sistemas de comunicación de red. Fue desarrollado por la ISO (International Standars Organization) como el primer paso hacia la estandarización internacional de los protocolos. Se compone de siete capas que representan las diferentes funciones de la comunicación de red, desde la transmisión física de datos hasta la presentación y la interacción con las aplicaciones de usuario. Estas capas se

organizan en una jerarquía, y cada una de ellas se encarga de funciones específicas sin necesidad de conocer los detalles de las capas inferiores o superiores. Cada capa se comunica solamente con su capa inferior y superior, permitiendo la implementación independiente de los servicios en cada una de ellas. De igual manera, cada capa es responsable de un grupo de servicios para el proceso de transmisión de la información entre dispositivos a través de protocolos de un nodo a otro [11]. Las capas del modelo OSI son:

1. Capa física: Contiene los elementos físicos de la red, se determinan las características físicas del cableado, codificación de señales, dispositivos, limitaciones, entre otros [11].
2. Capa de enlace: Su función principal es convertir un medio de transmisión en una línea libre de errores, enmascarando los errores reales para que la capa de red no sea capaz de visualizarlos. Para esto, el emisor divide los datos en tramas, para posteriormente transmitirlas de forma secuencial. De igual manera, en esta capa y en las capas superiores se realiza un proceso de control de flujo de datos para evitar que un transmisor rápido inunde de datos a un receptor lento [9].
3. Capa de red: Encargada del tráfico de la red, por lo que se asegura de entregar y direccionar los paquetes de datos.De igual manera, controla la operación de la subred, y determina la manera en que se encaminan los paquetes desde el origen hasta el destino [11].
4. Capa de transporte: Es la capa encargada de controlar el flujo de datos entre nodos de una manera eficiente. Se asegura de que los datos se entreguen con un correcto control de flujo y una detección de errores eficiente. De igual manera, determina el tipo de servicio que se debe proveer a la capa de sesión y a los usuarios de la red [11].

   "La capa de transporte es una verdadera capa de extremo a extremo; lleva los datos por toda la ruta desde el origen hasta el destino. En otras palabras, un programa en la máquina de origen lleva a cabo una conversación con un programa similar en la máquina de destino mediante el uso de los encabezados en los mensajes y los mensajes de control [9]

5. Capa de sesión: Es la capa encargada de establecer, mantener y finalizar las conexiones entre dos aplicaciones de diferentes dispositivos. En otras palabras, coordina la comunicación entre las aplicaciones en los extremos de la comunicación, lo que asegura que los datos se transmitan de una manera sincronizada y confiable [9].
6. Capa de presentación: También conocida como la cara o el traductor del modelo OSI, la capa de presentación se encarga de tomar los paquetes de la capa de aplicación para darles formato a los datos [11]. A diferencia de las capas anteriores, las cuales tienen como tarea principal mover bits de un lado a otro, esta capa se centra en la sintaxis y semántica de la información transmitida [9].
7. Capa de aplicación: Es la encargada de habilitar la interfaz que emplea el usuario en su computadora o dispositivo [11].

En la figura 8 se plasma cómo interactúa cada una de las capas en el modelo OSI.

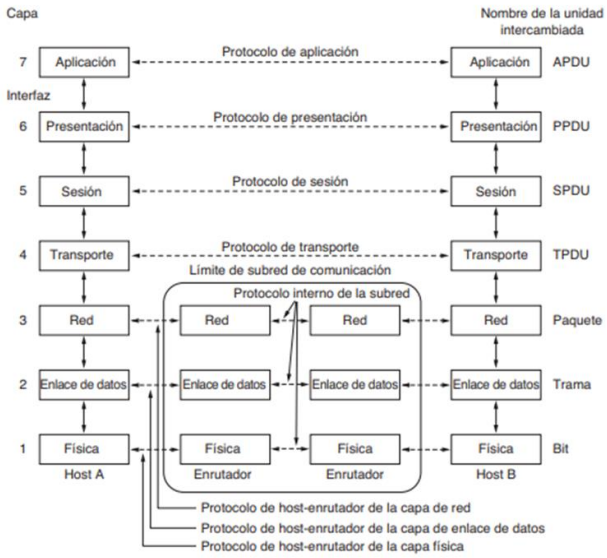


*Fig. 8. Capas del modelo OSI [9].*

## G. *Tipos de redes*

Las redes se clasifican en distintas categorías según su alcance geográfico y la cantidad de dispositivos que interconectan. Algunas de estas son:

- LAN (Local Area Network): Redes pequeñas en donde cada equipo está conectado al resto, se suele usar para oficinas, edificios, hogares, entre otros.
- MAN (Metropolitan Area Network): Redes diseñadas para cubrir zonas geográficas grandes pero un poco más limitadas que las WAN, como una ciudad o población determinada.
- WAN (Wide Area Network): Redes que cumplen un área geográfica extensa. Puede incluir países o incluso continentes.

En la figura 9 se visualiza una comparación gráfica entre cada uno de los tipos de red mencionados.


*Fig. 9. Tipos de red en [12].*

*H. VLAN*

Las VLAN (Virtual Local Area Network) permiten crear segmentos de redes físicas en múltiples redes lógicas. Se configuran mediante software en los switches y se comunican entre sí mediante enlaces troncales. De igual manera, cada VLAN funciona como una red independiente y tiene su propio espacio de direcciones IP y su propio dominio broadcast.

Primero, se generó cada VLAN con su respectivo nombre dentro de la base de datos en el switch router en la figura 10.

```
Rack#vlan database
Rack(vlan)#vlan 2 name DESARROLLO
VLAN 2 added:
    Name: DESARROLLO
Rack(vlan)#VLAN B?
% Unrecognized command
Rack(vlan)#vlan 3 name GERENCIA
VLAN 3 added:
    Name: GERENCIA
Rack(vlan)#vlan 4 name SISTEMASN
VLAN 4 added:
    Name: SISTEMASN
Rack(vlan)#vlan 5 name SERVIDORES
VLAN 5 added:
    Name: SERVIDORES
```

*Fig. 10. Creación de Vlans, elaboración propia.*

Al mostrar cada una se observa que cada una está por defecto configurada ethernet, con un tamaño máximo de 1500 bits. Lo correspondiente al departamento de Investigación se realizó después para evitar problemas con IPv6 en la figura 11.

```
Rack(vlan)#show
  VLAN ISL Id: 1
    Name: default
    Media Type: Ethernet
    VLAN 802.10 Id: 100001
    State: Operational
    MTU: 1500
    Translational Bridged VLAN: 1002
    Translational Bridged VLAN: 1003

  VLAN ISL Id: 2
    Name: DESARROLLO
    Media Type: Ethernet
    VLAN 802.10 Id: 100002
    State: Operational
    MTU: 1500

  VLAN ISL Id: 3
    Name: GERENCIA
    Media Type: Ethernet
    VLAN 802.10 Id: 100003
    State: Operational
    MTU: 1500

  VLAN ISL Id: 4
    Name: SISTEMASN
    Media Type: Ethernet
    VLAN 802.10 Id: 100004
    State: Operational
    MTU: 1500

  VLAN ISL Id: 5
    Name: SERVIDORES
    Media Type: Ethernet
    VLAN 802.10 Id: 100005
    State: Operational
    MTU: 1500
```

*Fig. 11. Comprobación creación de Vlans, elaboración propia.*

Luego, se configuraron simultáneamente múltiples interfaces de red dentro de un rango especificado, habilitando el acceso de los puertos y asignándole la VLAN correspondiente. Los últimos puertos fueron destinados a la creación de enlaces troncales, como se ve en la figura 12.

```
Rack#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Rack(config)#interface range fastEthernet 1/0 - 2
Rack(config-if-range)#switchport mode access
Rack(config-if-range)#switchport access vlan 2
Rack(config-if-range)#exit
Rack(config)#interface range fastEthernet 1/3 - 5
Rack(config-if-range)#switchport mode access
Rack(config-if-range)#switchport access vlan 3
Rack(config-if-range)#exit
Rack(config)#interface range fastEthernet 1/6 - 8
Rack(config-if-range)#switchport mode access
Rack(config-if-range)#switchport access vlan 4
Rack(config-if-range)#exit
Rack(config)#interface range fastEthernet 1/9 - 11
Rack(config-if-range)#switchport mode access
Rack(config-if-range)#switchport access vlan 5
Rack(config-if-range)#exit
Rack(config)#interface range fastEthernet 1/12 - 15
Rack(config-if-range)#switchport mode trunk
Rack(config-if-range)#switchport trunk encapsulation dot1q
Rack(config-if-range)#exit
```

*Fig. 12. Configuración Rack en GNS3, elaboración propia.*

La figura 13 muestra las VLAN creadas con su respectivo nombre, puertos y estado activo con el comando no shutdown.

```
RackP2N#show vlan-switch

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa1/15
2    DESARROLLO                       active    Fa1/0, Fa1/1, Fa1/2
3    GERENCIA                         active    Fa1/3, Fa1/4, Fa1/5
4    SISTEMASN                        active    Fa1/6, Fa1/7, Fa1/8
5    SERVIDORES                       active    Fa1/9, Fa1/10, Fa1/11
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        1002   1003
2    enet  100002     1500  -      -      -        -    -        0      0
3    enet  100003     1500  -      -      -        -    -        0      0
4    enet  100004     1500  -      -      -        -    -        0      0
5    enet  100005     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        1      1003
1003 tr    101003     1500  1005   0      -        -    srb      1      1002
1004 fdnet 101004     1500  -      -      1        ibm  -        0      0

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1005 trnet 101005     1500  -      -      1        ibm  -        0      0
```

*Fig. 13. Comprobación configuración Rack puertos access, elaboración propia.*

A partir de lo anterior, se muestran las interfaces troncales.

```
RackP2N#show interfaces trunk

Port      Mode         Encapsulation  Status        Native vlan
Fa1/12    on           802.1q         trunking      1
Fa1/13    on           802.1q         trunking      1
Fa1/14    on           802.1q         trunking      1

Port      Vlans allowed on trunk
Fa1/12    1-5,1002-1005
Fa1/13    1-5,1002-1005
Fa1/14    1-5,1002-1005

Port      Vlans allowed and active in management domain
Fa1/12    1-5
Fa1/13    1-5
Fa1/14    1-5

Port      Vlans in spanning tree forwarding state and not pruned
Fa1/12    1-5
Fa1/13    1-5
Fa1/14    1-5
```

*Fig. 14. Comprobación configuración Rack puertos troncales, elaboración propia.*

### *I. Ethernet*

Es un estándar de tecnología de red el cual define la forma en que los dispositivos de una LAN se comunican entre sí. Opera en la capa de enlace del modelo OSI (capa 2) y se encarga de la comunicación entre dispositivos, utilizando una variedad de protocolos para gestionar la transmisión de datos y la detección de colisiones [9].

### *J. Fast Ethernet*

Es una versión mejorada del estándar Ethernet, se desarrolló para satisfacer la creciente demanda de velocidad en redes locales. En comparación con Ethernet tradicional, el cual opera a velocidades de hasta 10Mbps, Fast Ethernet proporciona un gran aumento en la velocidad de transmisión de datos, ofreciendo velocidades de transmisión de hasta 100Mbps. Utiliza la misma arquitectura tradicional y el mismo formato de trama que Ethernet tradicional [9].

### *K. Direcciones MAC*

Las direcciones MAC (Media Access control) son identificadores únicos que se asignan a cada dispositivo de red a nivel de hardware. Estas direcciones se asignan con el fin de identificar de manera única cada dispositivo. Todos los dispositivos que cuenten con una tarjeta o adaptador de red tendrán una dirección MAC grabada en su hardware. Las direcciones MAC constan de 48 bits y a menudo se representan en formato hexadecimal [9].

### *L. Direcciones IP*

Las direcciones IP son un identificador asignado a cada dispositivo conectado a una red que utiliza el protocolo IP para la comunicación. Es una dirección lógica y se utiliza para identificar la ubicación de un dispositivo en una red. Se utiliza en la capa de red del modelo OSI y sirve para enrutar paquetes de datos a través de la red [9].

Pueden ser estáticas (asignadas manualmente) o dinámicas (asignadas automáticamente por un servidor DHCP). Las direcciones de IPv4 se conforman por 32 bits y se representan en decimal, mientras que las direcciones de IPv6 constan de 128 bits y se representan en formato hexadecimal [9].

### *M. Broadcast*

Es una forma de comunicación en la que un dispositivo envía datos a todos los demás dispositivos en la red. En otras palabras, cuando un dispositivo envía un mensaje de tipo broadcast, este se difunde a todos los nodos dentro de la red independientemente de sus direcciones MAC o IP. Generalmente se utilizan para la detección de dispositivos en la red o la configuración automática de direcciones IP. Al hablar de direcciones MAC su dirección es 'ffff.ffff.ffff', así que si un equipo envía un paquete de datos con esa dirección MAC como destino, este se enviará a todos los dispositivos en la red [9].

### *N. Topología de red*

La topología de red se define como el mapa físico/lógico de una red para la transmisión e intercambio de datos, en otras palabras, es la forma en la que está diseñada la red. El concepto de red se puede definir como un conjunto de nodos interconectados; teniendo en cuenta que un nodo hace referencia a cualquier dispositivo conectado a una red que cuenta con una dirección única y es capaz de enviar, recibir o retransmitir datos [11].

El éxito de una red está definido por una variedad de factores, como lo son: potencia, rendimiento, alta disponibilidad y libertad de conexión sin importar los elementos físicos y lógicos. Es importante destacar que no hay una solución única ni una verdad absoluta en una topología de red [11].

### *O. Enlaces ethernet*

Un enlace ethernet es la conexión física que utiliza tecnología ethernet para la comunicación entre dispositivos de red. Pueden utilizar diversos medios de transmisión. Su estándar es IEEE 802.3 [9].

### *P. Enlaces troncales*

Es un enlace de red que transporta datos entre dos dispositivos (generalmente switches o routers), y puede transportar simultáneamente. Este enlace permite la segmentación del tráfico y el paso de múltiples VLAN a través de un solo enlace. Se configura utilizando el protocolo IEEE802.1Q [9].

*Q. Enlaces redundantes*

Un enlace se refiere a la conexión física o lógica entre dos dispositivos de red que les permite comunicarse entre sí, además de permitir el intercambio de datos. Por lo tanto, los enlaces redundantes son conexiones adicionales entre dispositivos los cuales se usan para aumentar la disponibilidad de la red y asegurar que en caso de que un enlace falle, el tráfico de la red pueda seguir fluyendo a través de algún enlace alternativo [13].

No tener enlaces redundantes en una red puede abrir paso a varios problemas, como lo son:

- Puntos únicos de falla: Al no tener enlaces redundantes, se depende de un solo camino para la comunicación entre dispositivos, por lo que si ese camino falla, puede resultar en falta de conectividad en la red e interrupción del servicio [13].
- Rendimiento reducido: En caso de tener un alto volumen de tráfico, los enlaces congestionados pueden ralentizar el rendimiento de la red [13].
- Escalabilidad limitada: El no uso de enlaces redundantes resultará en dificultades para manejar un crecimiento futuro en el tráfico de datos o en el número de dispositivos conectados, lo que se traduce en limitaciones en su capacidad para adaptarse a nuevas demandas [13].

Sin embargo, el tener enlaces redundantes puede generar algunas consecuencias negativas, como el incremento de costos, consumo adicional de recursos de red, mayor complejidad en la topología y la posibilidad de congestión y bucles en la red. No obstante, los problemas de congestión de tráfico y los bucles de red se pueden corregir mediante la implementación de protocolos que garantizan la estabilidad y eficiencia de la red [13].

*R. STP*

STP (Spanning tree protocol) es un protocolo de red de capa 2 utilizado para evitar bucles dentro de una topología de red. Fue creado para evitar problemas que surgen cuando los dispositivos intercambian datos en una red LAN con rutas o enlaces redundantes, ya que en caso de que el flujo de tráfico no se monitoree adecuadamente, los datos pueden quedar atrapados en un bucle el cual gira alrededor de los segmentos de red, lo que afecta el rendimiento de esta e incluso puede hacer que se detenga el tráfico de datos. Un bucle ocurre cuando los datos viajan desde un origen a un destino a lo largo de rutas y enlaces redundantes, por lo que los datos comienzan a girar alrededor de los mismos caminos a manera de ciclo, resultando en una tormenta de transmisión [14].

Concretamente, STP tiene como objetivo principal monitorear todos los enlaces de red, identificando conexiones redundantes y desactivando los puertos que pueden provocar bucles. Dado esto, STP se utiliza cuando se desea incluir el uso de enlaces redundantes en la red, pero eliminar la probabilidad de bucles [14].

Para su correcto funcionamiento, STP se podría dividir en los siguientes pasos:

1. Elección de un puente raíz: Un puente raíz en una red es un puente que se elige el nodo central y punto de referencia para la configuración de la topología de red. Para esto, se le asigna una ID de puente única a cada puente de la red (suele estar compuesta por la dirección MAC del puente y un valor de prioridad). El puente con la ID más baja es aquel que se convierte en el puente de raíz de la red. Es importante destacar que cada VLAN debe tener su propio puente raíz, ya que cada VLAN es un dominio de difusión independiente [15].
2. Cálculo del camino más corto: Una vez se tenga elegido el puente de raíz, cada puente calcula el camino activo más corto hacia este. Para llevar a cabo este proceso, cada switch utiliza la información recopilada de las BPDU (Bridge Protocol Data Unit) intercambiadas entre los switches. Esta información incluye la ID del puente raíz, la distancia hacia él y el estado de los enlaces [15].
3. Desactivación de enlaces redundantes: Cuando ya se tienen determinados los caminos más cortos hacia el puente raíz, se desactivan selectivamente los enlaces redundantes para evitar la formación de bucles en la red. Para esto, los enlaces que no se encuentran en la ruta más corta hacia el puente raíz se ponen en estado de bloqueo, por lo que no se usan para el tráfico de datos [15].
4. Monitoreo y reconfiguración dinámica: Completado el proceso, STP monitorea constantemente la topología de la red para detectar cambios, como la eliminación o adición de enlaces. En ese caso, STP repite el proceso anteriormente mencionado y ajusta dinámicamente la configuración de los enlaces [15].

Para comprender mejor la manera en que funciona STP, se puede tomar como ejemplo la topología en la figura 5.

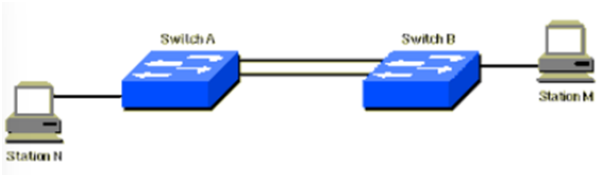


*Fig. 14. Topología con enlaces redundantes en [15].*

En dicha topología de red se cuenta con dos hosts (Station N y Station M) y dos switches (switch A y Switch B) y se cuenta con un enlace redundante entre el Switch A y el Switch B. En caso de que se desee enviar datos desde la estación M hasta la estación N, existe la posibilidad de que estos caigan en un bucle gracias a los enlaces redundantes. Es en este momento cuando STP interviene para que la topología de la red se vea lógicamente de la siguiente manera:

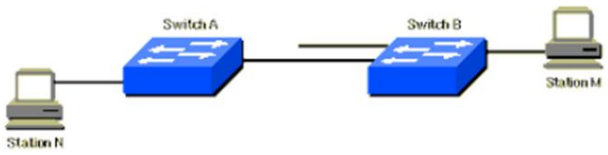


*Fig. 15. Vista lógica de topología con enlaces redundantes después de usar STP en [15].*

En este caso, STP se asegura de que haya un único enlace activo entre el switch A y el switch B, por lo que no hay riesgos de que se generen bucles.

La implementación de lo anterior se muestra a continuación en ambos routers, (implementación router 1 en figura 16 y router 2 en figura 17) donde se crea para cada VLAN con sus correspondientes prioridades. Luego, se muestra cada prueba (figura 18 a 20 router 1 y figura 21 a 23 router 2).

```
RackP1N#p}
RackP1N#show spanning-tree brief

VLAN1
  Spanning tree enabled protocol ieee
  Root ID    Priority    100
             Address     cc03.1724.0001
             Cost        19
             Port        54 (FastEthernet1/13)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    200
             Address     cc04.1844.0001
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/12     128.53   128    19 FWD    19   200 cc04.1844.0001 128.53
FastEthernet1/13     128.54   128    19 FWD     0   100 cc03.1724.0001 128.54
FastEthernet1/15     128.56   128    19 FWD    19   200 cc04.1844.0001 128.56


VLAN2
  Spanning tree enabled protocol ieee
  Root ID    Priority    300
             Address     cc03.1724.0002
             Cost        19
             Port        54 (FastEthernet1/13)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    400
             Address     cc04.1844.0000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/0      128.41   128    19 FWD    19   400 cc04.1844.0000 128.41
FastEthernet1/1      128.42   128    19 FWD    19   400 cc04.1844.0000 128.42
FastEthernet1/12     128.53   128    19 FWD    19   400 cc04.1844.0000 128.53
FastEthernet1/13     128.54   128    19 FWD     0   300 cc03.1724.0002 128.54
FastEthernet1/15     128.56   128    19 FWD    19   400 cc04.1844.0000 128.56
```

*Fig. 16. Priorización protocolo spanning-tree router 1, elaboración propia.*

```
RackP2N#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
RackP2N(config)#spanning-tree vlan 1 100
                                   ^
% Invalid input detected at '^' marker.

RackP2N(config)#spanning-tree vlan 1 priority 100
RackP2N(config)#spanning-tree vlan 2 priority 300
RackP2N(config)#spanning-tree vlan 3 priority 400
RackP2N(config)#spanning-tree vlan 5 priority 500
RackP2N(config)#spanning-tree vlan 4 priority 500
RackP2N(config)#spanning-tree vlan 5 priority 200
RackP2N(config)#EXIT
RackP2N#WRITE
*Mar  1 00:09:34.983: %SYS-5-CONFIG_I: Configured from console by console
RackP2N#WRITE
Building configuration...
```

*Fig. 17. Priorización protocolo spanning-tree router 2, elaboración propia.*

```
RackP2N#show spanning-tree brief

VLAN1
  Spanning tree enabled protocol ieee
  Root ID    Priority    100
             Address     cc03.4b44.0002
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    100
             Address     cc03.4b44.0002
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/12     128.53   128    19 FWD     0   100 cc03.4b44.0002 128.53
FastEthernet1/13     128.54   128    19 FWD     0   100 cc03.4b44.0002 128.54
FastEthernet1/14     128.55   128    19 FWD     0   100 cc03.4b44.0002 128.55


VLAN2
  Spanning tree enabled protocol ieee
  Root ID    Priority    300
             Address     cc03.4b44.0003
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    300
             Address     cc03.4b44.0003
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/12     128.53   128    19 FWD     0   300 cc03.4b44.0003 128.53
FastEthernet1/13     128.54   128    19 FWD     0   300 cc03.4b44.0003 128.54
FastEthernet1/14     128.55   128    19 FWD     0   300 cc03.4b44.0003 128.55
```

*Fig. 18. Spanning-tree en un switch de prioridad 2, elaboración propia.*

```
VLAN3
  Spanning tree enabled protocol ieee
  Root ID    Priority    400
             Address     cc03.1724.0003
             Cost        19
             Port        54 (FastEthernet1/13)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    500
             Address     cc04.1844.0002
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/12     128.53   128    19 FWD    19   500 cc04.1844.0002 128.53
FastEthernet1/13     128.54   128    19 FWD     0   400 cc03.1724.0003 128.54
FastEthernet1/15     128.56   128    19 FWD    19   500 cc04.1844.0002 128.56


VLAN4
  Spanning tree enabled protocol ieee
  Root ID    Priority    500
             Address     cc03.1724.0000
             Cost        19
             Port        54 (FastEthernet1/13)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    600
             Address     cc04.1844.0003
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/12     128.53   128    19 FWD    19   600 cc04.1844.0003 128.53
FastEthernet1/13     128.54   128    19 FWD     0   500 cc03.1724.0000 128.54
FastEthernet1/15     128.56   128    19 FWD    19   600 cc04.1844.0003 128.56
```

*Fig. 19. Spanning-tree en un switch de prioridad 2, elaboración propia.*

```
VLAN5
  Spanning tree enabled protocol ieee
  Root ID    Priority    200
             Address     cc03.1724.0004
             Cost        19
             Port        54 (FastEthernet1/13)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    300
             Address     cc04.1844.0004
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/12     128.53   128    19 FWD    19   300 cc04.1844.0004 128.53
FastEthernet1/13     128.54   128    19 FWD     0   200 cc03.1724.0004 128.54
FastEthernet1/15     128.56   128    19 FWD    19   300 cc04.1844.0004 128.56
```

*Fig. 20. Spanning-tree en un switch de prioridad 1, elaboración propia.*

```
RackP2N#show spanning-tree brief

VLAN1
  Spanning tree enabled protocol ieee
  Root ID    Priority    100
             Address     cc03.4b44.0002
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    100
             Address     cc03.4b44.0002
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/12     128.53   128    19 FWD     0   100 cc03.4b44.0002 128.53
FastEthernet1/13     128.54   128    19 FWD     0   100 cc03.4b44.0002 128.54
FastEthernet1/14     128.55   128    19 FWD     0   100 cc03.4b44.0002 128.55


VLAN2
  Spanning tree enabled protocol ieee
  Root ID    Priority    300
             Address     cc03.4b44.0003
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    300
             Address     cc03.4b44.0003
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/12     128.53   128    19 FWD     0   300 cc03.4b44.0003 128.53
FastEthernet1/13     128.54   128    19 FWD     0   300 cc03.4b44.0003 128.54
FastEthernet1/14     128.55   128    19 FWD     0   300 cc03.4b44.0003 128.55
```

*Fig. 21. Spanning-tree en un switch de prioridad 1, elaboración propia.*

```
VLAN3
  Spanning tree enabled protocol ieee
  Root ID    Priority    400
             Address     cc03.4b44.0000
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    400
             Address     cc03.4b44.0000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/3      128.44   128    19 FWD     0   400 cc03.4b44.0000 128.44
FastEthernet1/4      128.45   128    19 FWD     0   400 cc03.4b44.0000 128.45
FastEthernet1/12     128.53   128    19 FWD     0   400 cc03.4b44.0000 128.53
FastEthernet1/13     128.54   128    19 FWD     0   400 cc03.4b44.0000 128.54
FastEthernet1/14     128.55   128    19 FWD     0   400 cc03.4b44.0000 128.55


VLAN4
  Spanning tree enabled protocol ieee
  Root ID    Priority    500
             Address     cc03.4b44.0001
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    500
             Address     cc03.4b44.0001
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300
```

*Fig. 22. Spanning-tree en un switch de prioridad 1, elaboración propia.*

```
VLAN5
  Spanning tree enabled protocol ieee
  Root ID    Priority    200
             Address     cc03.4b44.0004
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    200
             Address     cc03.4b44.0004
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                Designated
Name                 Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
-------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet1/12     128.53   128    19 FWD     0   200 cc03.4b44.0004 128.53
FastEthernet1/13     128.54   128    19 FWD     0   200 cc03.4b44.0004 128.54
FastEthernet1/14     128.55   128    19 FWD     0   200 cc03.4b44.0004 128.55
```

*Fig. 23. Spanning-tree en un switch de prioridad 1, elaboración propia.*

### *S. ARP*

ARP (Address Resolution Protocol) es un protocolo de comunicación que se usa para asociar direcciones IP con direcciones MAC. Dicho en otras palabras, su función principal es encontrar la dirección MAC de un dispositivo, conociendo su dirección IP [9].

Su funcionamiento básico implica dos operaciones:

1. Solicitud ARP (ARP Request): Si un dispositivo necesita comunicarse con otro dispositivo de la misma red local y no tiene conocimiento de la dirección MAC de destino, se envía una solicitud de ARP, la cual contiene la dirección IP del dispositivo de destino. Dicha solicitud se transmite a todas las máquinas de la red utilizando una trama broadcast [9].
2. Respuesta ARP (ARP Reply): El dispositivo que cuente con la dirección IP solicitada responde con su dirección MAC. Esta respuesta contiene la dirección IP correspondiente y la dirección MAC del dispositivo [9].

Una vez completado el proceso, el emisor recibe la respuesta ARP y asocia la dirección MAC con la dirección IP correspondiente. Esto se almacena en una tabla caché ARP para futuras ocasiones.

### *T. IPv4*

IPv4 o Intenernet Protocol versión 4 es un protocolo de internet usado para identificar y localizar dispositivos en una red mediante direcciones IP únicas. Funciona en la capa de red del modelo OSI y es esencial en el proceso de enrutamiento de paquetes de datos por medio de internet y redes IP.

Su funcionamiento consiste en que cada dispositivo conectado a una red IP tiene asignada una dirección IPv4 única, la cual está compuesta por 32 bits, divididos en 4 octetos (cada una de 1 byte). Estas direcciones se escriben en formato decimal y son usadas para identificar cada dispositivo de red de una forma única, facilitando el intercambio de datos.

Cuenta con un encabezado IPv4, el cual es una estructura de datos que se encuentra al principio de cada paquete y tiene un tamaño fijo de 20 bytes (aunque puede ampliarse a 60) distribuidos de la siguiente manera:

- Versión (4 bits): Indica la versión de IP.
- Tamaño del encabezado (4 bits): Indica la longitud del encabezado en palabras de 32 bits. El valor mínimo es de 5 palabras (160 bits o 20 bytes) y el valor máximo es de 15 palabras (480 bits o 60 bytes).
- Tipo de servicio (8 bits): Está pensado para especificar el nivel de prioridad del paquete, aunque no es muy usado en la actualidad.
- Longitud total (16 bits): Indica la longitud total del paquete.
- Identificador (16 bits): Un identificador único para el paquete. Este se usa en caso de que el paquete necesite ser fragmentado.
- Bandera (3 bits): Contiene 3 banderas (un bit para cada una). Una de ellas establece si el paquete no debe estar fragmentado, la segunda si el paquete es un fragmento de un paquete más grande, y la tercera indica la posición del fragmento dentro del paquete original.
- Tiempo de vida (TTL) (8 bits): Un contador que decrementa en uno por cada enrutador que atraviesa el paquete, en caso de que llegue a cero el paquete se descarta. Esto se usa para evitar que los paquetes circulan indefinidamente por la red en caso de errores de enrutamiento o bucles infinitos.
- Protocolo (8 bits): indica el protocolo de transporte que se utiliza.
- Suma de comprobación (16 bits): Utilizada para encontrar errores en el encabezado.
- Dirección IP de origen (32 bits): La dirección IP del dispositivo que envió el paquete.
- Dirección IP de destino (32 bits): La dirección IP del dispositivo al que se envía el paquete.

En la figura 24 se puede observar el encabezado completo para IPv4.

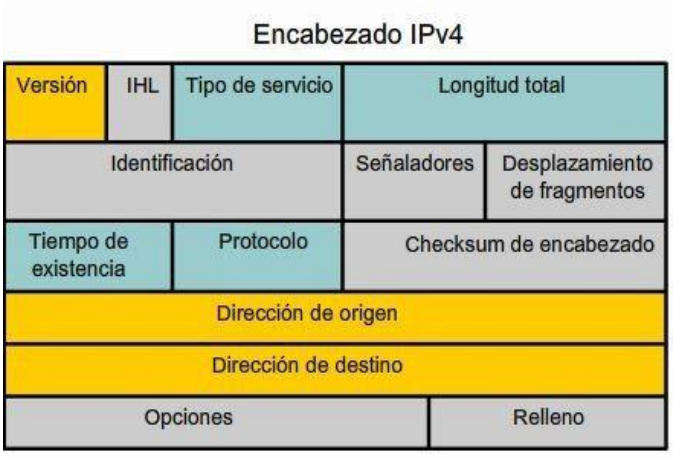


*Fig. 24. Encabezado de IPv4 en [16].*

*U. IPv6*

Es la última versión del protocolo de internet (IP) y se creó para solucionar limitaciones de su versión anterior (IPv4), principalmente el agotamiento de direcciones IP. Por esto mismo pasó de utilizar direcciones de 32 bits en su versión anterior a direcciones de 128 bits en forma hexadecimal.

Uno de los cambios con respecto a IPv4 es la estructura de su encabezado, ya que en IPv6 se cuenta con un tamaño fijo de 40 bytes, los cuales se distribuyen de la siguiente manera:

- Versión (4 bits): Indica la versión de IP. Para IPv6 se debe establecer en 0110.
- Clase de tráfico (8 bits): Define la prioridad del paquete.
- Etiqueta de flujo (20 bits): Se utiliza para identificar y clasificar flujos de tráfico específicos.
- Identificador de extensión (16 bits): Indica si el paquete contiene extensiones de encabezado.
- Longitud de la carga útil (16 bits): Indica la longitud de los datos del paquete.
- Siguiente encabezado (8 bits): Usado para identificar el protocolo de la capa superior que procesa el paquete.
- Salto (8 bits): Similar al TTL de IPv4, pero ya no se usa para evitar bucles sino para optimizar el enrutamiento. El valor inicial de este campo es 0 e incrementa en uno por cada enrutador que procesa el paquete. Si el valor alcanza el límite (63) el paquete se descarta.
- Dirección IP de origen (128 bits): La dirección IP del dispositivo que envió el paquete.
- Dirección IP de destino (128 bits): La dirección IP del dispositivo al que se envía el paquete.

En la figura 28 se puede observar el encabezado completo para IPv6 y realizar una comparación en cuanto al encabezado de IPv4.

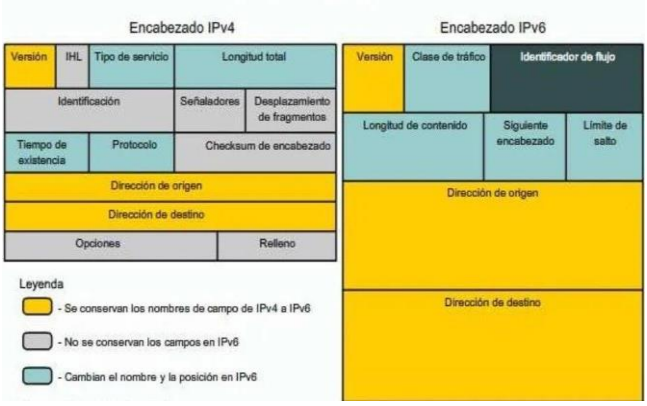


*Fig. 28. Encabezado de IPv4 en comparación al encabezado de IPv6 en [17].*

En la figura 29 se puede observar que como primer paso se inicia la sesión de configuración en el modo de configuración terminal, posteriormente se accede al modo de configuración de la interfaz FastEthernet respectiva y se asignan ambas direcciones IP.

```
RD#
RD#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
RD(config)#interface fastEthernet 0/1.2
RD(config-subif)#encapsulation mode access
                            ^
% Invalid input detected at '^' marker.

RD(config-subif)#encapsulation dot1q 2
RD(config-subif)#ip address 192.168.5.1 255.255.255.192
RD(config-subif)#ipv6 address FD00:3::1/64
RD(config-subif)#exit
RD(config)#interface fastEthernet 0/1.3
RD(config-subif)#encapsulation dot1q 3
RD(config-subif)#ip address 192.168.5.65 26B?
% Unrecognized command
RD(config-subif)#ip address 192.168.5.65 255.255.255.224
RD(config-subif)#ipv6 address FD00:4::1/64
RD(config-subif)#exit
RD(config)#interface fastEthernet 0/1.4
RD(config-subif)#encapsulation dot1q 4
RD(config-subif)#ip address 192.168.5.97 255.255.255.224
RD(config-subif)#ipv6 address FD00:5::1/64
RD(config-subif)#exit
RD(config)#interface fastEthernet 0/1.5
RD(config-subif)#encapsulation dot1q 5
RD(config-subif)#ip address 192.168.5.129 255.255.255.240
RD(config-subif)#ipv6 address FD00:6::1/64
RD(config-subif)#exit
RD(config)#interface fastEthernet 0/1
RD(config-if)#no shutdown
RD(config-if)#e
*Mar  1 00:09:56.363: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
*Mar  1 00:09:57.363: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/1, changed state to up
RD(config-if)#exit
RD(config)#exit
```

*Fig. 29. Configuración de las direcciones IPv4 e IPv6 del router en GNS3, elaboración propia.*

```
RD#show i
*Mar  1 00:10:04.475: %SYS-5-CONFIG_I: Configured from console by console
RD#show ip interface brief
Interface                  IP-Address      OK? Method Status                Prot
ocol
FastEthernet0/0            unassigned      YES unset  administratively down down

FastEthernet0/1            unassigned      YES unset  up                    up

FastEthernet0/1.2          192.168.5.1     YES manual up                    up

FastEthernet0/1.3          192.168.5.65    YES manual up                    up

FastEthernet0/1.4          192.168.5.97    YES manual up                    up

FastEthernet0/1.5          192.168.5.129   YES manual up                    up

FastEthernet1/0            unassigned      YES unset  administratively down down

RD#show ipv6 interface brief
FastEthernet0/0            [administratively down/down]
FastEthernet0/1            [up/up]
FastEthernet0/1.2          [up/up]
    FE80::CE02:59FF:FEA8:1
    FD00:3::1
FastEthernet0/1.3          [up/up]
    FE80::CE02:59FF:FEA8:1
    FD00:4::1
FastEthernet0/1.4          [up/up]
    FE80::CE02:59FF:FEA8:1
    FD00:5::1
FastEthernet0/1.5          [up/up]
    FE80::CE02:59FF:FEA8:1
    FD00:6::1
FastEthernet1/0            [administratively down/down]
RD#
```

*Fig. 30. Comprobación de direcciones IPv4 e IPv6 para router, elaboración propia.*

```
RD#
RD#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
RD(config)#interface fastEthernet 1/0
RD(config-if)#ip address 172.16.1.1 255.255.255.0
RD(config-if)#ipv6 address FD00:7::1/64
RD(config-if)#no shutdown
RD(config-if)#exit
RD(config)#
*Mar  1 00:18:34.535: %LINK-3-UPDOWN: Interface FastEthernet1/0, changed state t
o up
*Mar  1 00:18:35.535: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et1/0, changed state to up
RD(config)#
RD(config)#ip route 192.168.5.144 255.255.255.240 172.16.1.2
RD(config)#ip route 192.168.5.160 255.255.255.240 172.16.1.2
RD(config)#ipv6 route FD00:1::/64 FD00:7::2
RD(config)#ipv6 route FD00:2::/64 FD00:7::2
RD(config)#
RD(config)#ipv6 unicast-routing
RD(config)#
RD(config)#exit
RD#wri
*Mar  1 00:31:22.115: %SYS-5-CONFIG_I: Configured from console by console
RD#write
Building configuration...
[OK]
RD#
```

*Fig. 31. Enrutamiento IPv4 e IPv6 para router, elaboración propia.*

```
RD#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.168.16.2 to network 0.0.0.0

     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.1.0 is directly connected, FastEthernet1/0
     192.168.5.0/24 is variably subnetted, 6 subnets, 3 masks
C       192.168.5.96/27 is directly connected, FastEthernet0/1.4
C       192.168.5.64/27 is directly connected, FastEthernet0/1.3
C       192.168.5.0/26 is directly connected, FastEthernet0/1.2
S       192.168.5.160/28 [1/0] via 172.16.1.2
C       192.168.5.128/28 is directly connected, FastEthernet0/1.5
S       192.168.5.144/28 [1/0] via 172.16.1.2
C    192.168.16.0/24 is directly connected, FastEthernet0/0
S*   0.0.0.0/0 [254/0] via 192.168.16.2
```

*Fig. 32. Comprobación enrutamiento IPv4 para router, elaboración propia.*

```
RD#show ipv6 route
IPv6 Routing Table - 14 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
S   FD00:1::/64 [1/0]
     via FD00:7::2
S   FD00:2::/64 [1/0]
     via FD00:7::2
C   FD00:3::/64 [0/0]
     via ::, FastEthernet0/1.2
L   FD00:3::1/128 [0/0]
     via ::, FastEthernet0/1.2
C   FD00:4::/64 [0/0]
     via ::, FastEthernet0/1.3
L   FD00:4::1/128 [0/0]
     via ::, FastEthernet0/1.3
C   FD00:5::/64 [0/0]
     via ::, FastEthernet0/1.4
L   FD00:5::1/128 [0/0]
     via ::, FastEthernet0/1.4
C   FD00:6::/64 [0/0]
     via ::, FastEthernet0/1.5
L   FD00:6::1/128 [0/0]
     via ::, FastEthernet0/1.5
C   FD00:7::/64 [0/0]
     via ::, FastEthernet1/0
L   FD00:7::1/128 [0/0]
     via ::, FastEthernet1/0
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
     via ::, Null0
RD#
```

*Fig. 33. Comprobación enrutamiento IPv6 para router, elaboración propia.*

## III. INSTALACIÓN Y CONFIGURACIÓN DE SERVIDORES

En esta sección se ofrecen las bases teóricas, proceso de instalación, configuración de los servidores WEB, DNS, LDAP y DHCP y sus respectivas pruebas.

### *A. WEB (HTTP) y LDAP*

Los servidores web, también conocidos como servidores HTTP que significa Protocolo de Transferencia de Hipertexto (en inglés, Hypertext Transfer Protocol), son fundamentales para la disponibilidad constante y segura de contenidos en Internet. Según IONOS, "Los servidores web sirven para almacenar contenidos de Internet y facilitar su disponibilidad de forma constante y segura" [1]. Estos servidores son responsables de enviar los componentes individuales de una página web directamente a los ordenadores de los usuarios cuando estos acceden a la página.

De esta forma, tanto grandes empresas como pequeños proyectos, dentro de los que se incluye el presente prototipo, suelen recurrir a servidores web, ya sea propios o alquilados a proveedores de alojamiento web, para gestionar sus contenidos. Cabe resaltar que un servidor web ofrece una flexibilidad excepcional, permitiéndote ajustar y personalizar el entorno de servidor según las necesidades específicas de un proyecto. Esta capacidad de adaptación es particularmente útil en un entorno en constante evolución, donde los requisitos de rendimiento y las demandas del mercado pueden cambiar rápidamente.

Según Dongee, "Un servidor web ofrece una flexibilidad excepcional, permitiéndote ajustar y personalizar tu entorno de servidor según las necesidades específicas de tu proyecto" [2]. Además, esta flexibilidad permite implementar nuevas tecnologías y actualizar la infraestructura de manera eficiente con desventajas que resultan irrelevantes dado el tamaño del proyecto como la dependencia de un proveedor externo y costos adicionales.

Asimismo, la capacidad de escalabilidad de un servidor permite manejar una carga de trabajo intensiva. Un servidor robusto y escalable puede distribuir eficientemente la carga de trabajo entre múltiples servidores, dando una experiencia fluida, incluso durante los picos de tráfico más altos, lo que permite a su vez garantizar la disponibilidad y la fiabilidad del sitio en todo momento.

Por otro lado, LDAP (Protocolo Ligero de Acceso a Directorios) es un protocolo de software que permite a cualquier persona localizar datos sobre organizaciones, individuos y otros recursos, como archivos y dispositivos, en una red, ya sea en Internet público o en una intranet corporativa. De acuerdo con TechTarget, LDAP es una versión "ligera" del Protocolo de Acceso a Directorios (DAP), que forma parte de X.500, un estándar para servicios de directorio en una red. Se considera que LDAP es ligero porque utiliza una cantidad menor de código que otros protocolos [3].

En una red, un directorio indica al usuario dónde se encuentra algo. En las redes TCP/IP, incluida Internet, el sistema de nombres de dominio (DNS), el cual se abordará en la siguiente subsección, es el sistema de directorios utilizado para relacionar el nombre de dominio con una dirección de red específica, que es una ubicación única en la red. Sin embargo, el usuario puede no conocer el nombre de dominio. LDAP permite a un usuario buscar a una persona sin saber dónde está ubicada, aunque información adicional ayudará con la búsqueda.

Siguiendo la investigación en el libro Understanding and Deploying LDAP Directory Services [4], el modelo de información LDAP define los tipos de datos y las unidades básicas de información que se pueden almacenar en su directorio. En otras palabras, el modelo de información LDAP describe los bloques de construcción que se pueden utilizar para crear un directorio.

Las entradas, atributos y valores son las unidades básicas de información en el directorio. Por lo general, la información en una entrada describe un objeto del mundo real, como una persona, pero el modelo no requiere esto. En un directorio típico, se encuentran miles de entradas que corresponden a personas, departamentos, servidores, impresoras y otros objetos del mundo real en la organización servida por el directorio. Cada entrada de directorio tiene un nombre distinguido (DN); por ejemplo, la organización mostrada en la figura 34 tiene el DN dc=example, dc=com. Para el presente documento, se abordará una aproximación similar para la división de departamentos.

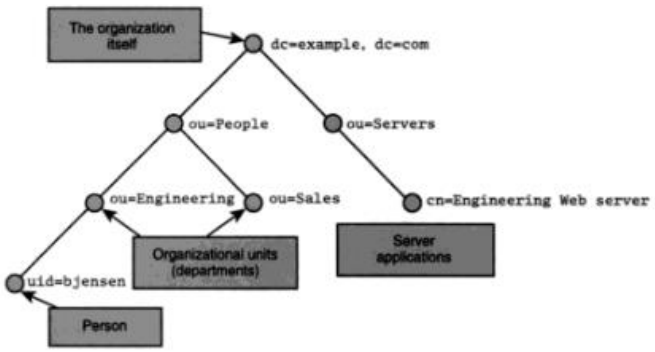


*Fig. 34. Sección de directorio típico LDAP en [4].*

Ahora bien, para implementar ambos servidores, WEB y LDAP, se usó Ubuntu Desktop, el cual despliega plug-ins para Apache en cuanto a servidor WEB y para LDAP, para el servidor del mismo nombre.

Primeramente, el libro INGE CUC afirma que “Apache es el servidor Web con mayor presencia en el mercado mundial” [5]. Esto se debe a su sencilla configuración, a pesar del problema en aumento que constituye la seguridad en los servidores HTTP. A continuación, se presenta el proceso de instalación y configuración de ambos servidores, el cual comienza con la instalación y configuración de Apache, la verificación del servidor, desarrollo de código HTML, activación de localhost, firewall y comprobación final.

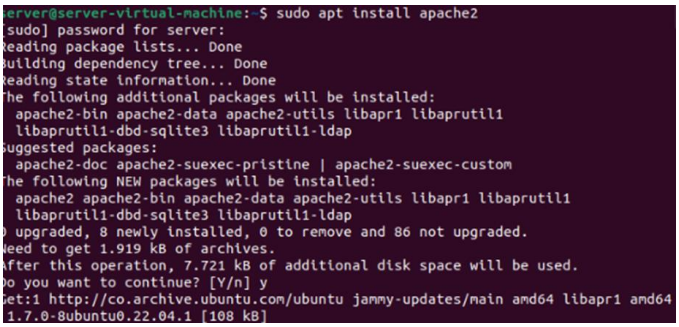


*Fig. 35. Actualización e instalación de APACHE WEB SERVER, elaboración propia.*

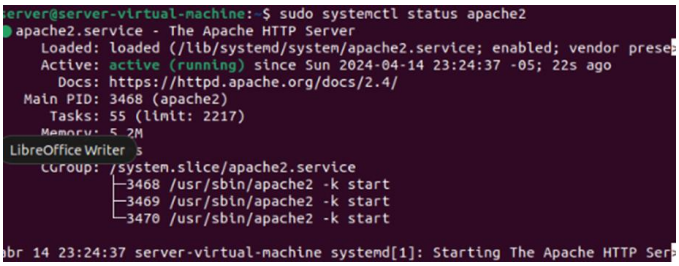


*Fig. 36. Verificacion del APACHE WEB SERVER activo, elaboración propia.*

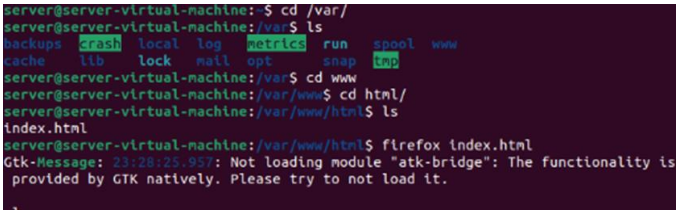


*Fig. 37. Desarrollo código HTML, elaboración propia.*


*Fig. 38. Activar el localhost, elaboración propia.*

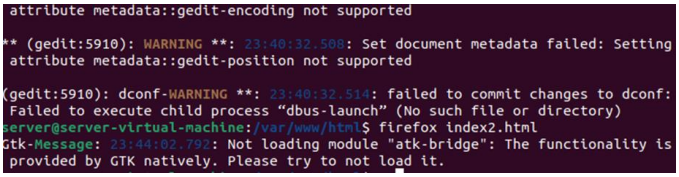


*Fig. 39. Abrir firefox con el index2.html, elaboración propia*

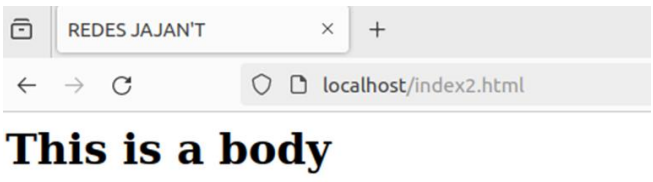


*Fig. 40. Servidor WEB, elaboración propia.*

En cuanto a la implementación de LDAP, en la configuración

## B. *DNS Y DHCP*

Como se mencionó anteriormente, un servidor DNS (Domain Name System) es aquel que "traduce los nombres de dominios aptos para lectura humana (por ejemplo, www.amazon.com) a direcciones IP aptas para lectura por parte de máquinas (por ejemplo, 192.0.2.44)" [6]. Amazon Web Service proporciona el siguiente ejemplo de cómo DNS dirige tráfico hacia su aplicación web.

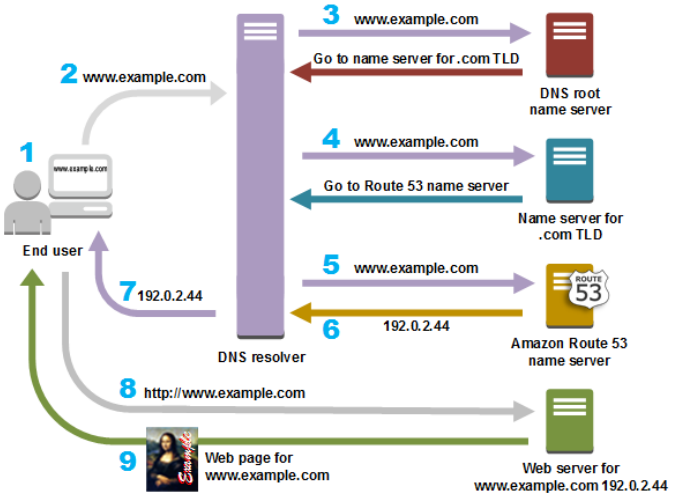


*Fig. 41. Ejemplo de DNS en [6].*

En este escenario, cuando un usuario abre un navegador web e ingresa la dirección www.ejemplo.com en la barra de direcciones, se inicia un proceso de resolución de DNS. La solicitud se envía al solucionador de DNS del proveedor de servicios de Internet (ISP) del usuario, que luego la vuelve a enviar a un servidor de nombres de raíz de DNS. Posteriormente, la solicitud se envía a un servidor de nombres de dominio de nivel superior (TLD) para los dominios .com, que responde con los servidores de nombres asociados al dominio ejemplo.com.

El solucionador de DNS del ISP elige uno de estos servidores y envía la solicitud allí. Este servidor busca en la zona alojada de ejemplo.com el registro de www.ejemplo.com y devuelve la dirección IP asociada, como 192.0.2.44. Esta dirección IP se devuelve al solucionador de DNS del ISP, que a su vez la proporciona al navegador web del usuario. El navegador envía entonces una solicitud a la dirección IP obtenida, donde se encuentra el contenido buscado, como un servidor web en una instancia de Amazon EC2 o un bucket de Amazon S3 configurado como un punto de enlace de sitio web. Finalmente, el servidor web en la dirección IP devuelve la página web al navegador del usuario para su visualización.

Por lo que se refiere a servidor DHCP, Dynamic Host Configuration Protocol, es un protocolo de gestión de red

utilizado para asignar dinámicamente una dirección IP a cualquier dispositivo o nodo en una red, para que pueda comunicarse utilizando IP. Nuevamente, TechTarget señala que este automatiza y gestiona de manera centralizada estas configuraciones en lugar de requerir que los administradores de red asignen manualmente direcciones IP a todos los dispositivos de la red, e igual que HTTP, puede implementarse en redes locales pequeñas, así como en redes empresariales grandes [7].

Así pues, DHCP funciona en la capa de aplicación del conjunto de protocolos TCP/IP. Asigna dinámicamente direcciones IP e información de configuración TCP/IP. Esta información incluye la máscara de subred, la dirección IP de la puerta de enlace predeterminada y las direcciones del sistema de nombres de dominio (DNS). En ella, los clientes envían una solicitud al servidor DHCP cada vez que se conectan a una red.

Los clientes configurados con DHCP transmiten una solicitud al servidor DHCP y solicitan información de configuración de red para la red local a la que están conectados. Un cliente típicamente transmite una consulta para esta información inmediatamente después de arrancar. El servidor DHCP responde a la solicitud del cliente proporcionando información de configuración IP previamente especificada por un administrador de red. Esto incluye una dirección IP específica, así como un período de tiempo, también llamado arrendamiento, durante el cual la asignación es válida.

De esta forma, al refrescar una asignación de dirección, un cliente DHCP solicita los mismos parámetros, pero el servidor DHCP puede asignar una nueva dirección IP según las políticas establecidas por los administradores. Los clientes DHCP también pueden ser configurados en una interfaz Ethernet.

El servidor DHCP administra un registro de todas las direcciones IP que asigna a los nodos de red. Si un nodo se reubica en la red, el servidor lo identifica utilizando su dirección de control de acceso a medios (MAC), lo que evita la configuración accidental de múltiples dispositivos con la misma dirección IP. Configurar un servidor DHCP también requiere la creación de un archivo de configuración, que almacena información de red para clientes.

Es de vital importancia decir que DHCP no es un protocolo enrutado ni seguro. Está limitado a una red de área local específica, lo que significa que un solo servidor DHCP por LAN es adecuado, o dos servidores para su uso en caso de un fallo. Redes más grandes pueden tener una red de área amplia (WAN) que contiene múltiples ubicaciones individuales. Dependiendo de las conexiones entre estos puntos y el número de clientes en cada ubicación, se pueden configurar múltiples servidores DHCP para manejar la distribución de direcciones.

Además, este carece de cualquier mecanismo incorporado que permita a clientes y servidores autenticarse mutuamente. Ambos son vulnerables a la suplantación, donde un ordenador finge ser otro, y al ataque, donde clientes falsos pueden agotar el grupo de direcciones IP de un servidor DHCP.

Similar a la subsección A, se presentarán los detalles de implementación de ambos servidores a partir de la interfaz en común que proporciona Ubuntu Server.

En la figura 42 se puede observar que se inicia con la creación de pools de desarrollo para DHCP y DNS.

```
RD#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
RD(config)#ip dhcp pool DESAROLLO
RD(dhcp-config)#network 192.168.5.0 255.255.255.192
RD(dhcp-config)#default-router 192.168.5.1
RD(dhcp-config)#dns-server 8.8.8.8
RD(dhcp-config)#exit
```

*Fig. 42. Creación pool DESARROLLO para DHCP Y DNS, elaboración propia.*

En la figura 43 se puede visualizar que se inicia con la creación de pools de las VLANs DHCP y DNS.

```
RD#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
RD(config)#ip dhcp pool GERENCIA
RD(dhcp-config)#network 192.168.5.64 255.255.255.224
RD(dhcp-config)#default-router 192.168.5.65
RD(dhcp-config)#dns-server 8.8.8.8
RD(dhcp-config)#exit
RD(config)#ip dhcp pool SISTEMASN
RD(dhcp-config)#network 192.168.5.96 255.255.255.224
RD(dhcp-config)#default-router 192.168.5.97
RD(dhcp-config)#dns-server 8.8.8.8
RD(dhcp-config)#exit
RD(config)#ip dhcp pool SERVIDORES
RD(dhcp-config)#network 192.168.5.128 255.255.255.240
RD(dhcp-config)#default-router 192.168.5.129
RD(dhcp-config)#dns-server 8.8.8.8
RD(dhcp-config)#EXIT
RD(config)#do write
Building configuration...
[OK]
RD(config)#
```

*Fig. 43. Creación pools de las VLANs para DHCP Y DNS, elaboración propia.*

En la figura 44 se puede observar que se inicia con la creación de pools de investigación para DHCP y DNS.

```
IPv6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
IPv6(config)#ip dhcp pool INV1
IPv6(dhcp-config)#network 192.168.5.144 255.255.255.240
IPv6(dhcp-config)#default-router 192.168.5.145
IPv6(dhcp-config)#dns-server 8.8.8.8
IPv6(dhcp-config)#exit
IPv6(config)#ip dhcp pool INV2
IPv6(dhcp-config)#network 192.168.5.160 255.255.255.240
IPv6(dhcp-config)#default-router 192.168.5.161
IPv6(dhcp-config)#dns-server 8.8.8.8
IPv6(dhcp-config)#exit
IPv6(config)#do write
Building configuration...
```

*Fig. 44. Creación pools de las VLANs de Investigación para DHCP Y DNS, elaboración propia.*

## IV. ESCENARIO DE PRUEBAS

Para iniciar, se empleó el algoritmo VLSM (variable length subnet mask) para calcular el tamaño de subredes de acuerdo con el número de host y la cantidad de subredes. Se presentan seguidamente imágenes con este proceso detallado.

Primero, se analiza la distribución física para determinar el número de subredes y hosts por cada una. Luego, se determina el número de bits sumando 2 al número de hosts para incluir la dirección de red y dirección broadcast en IPv4.

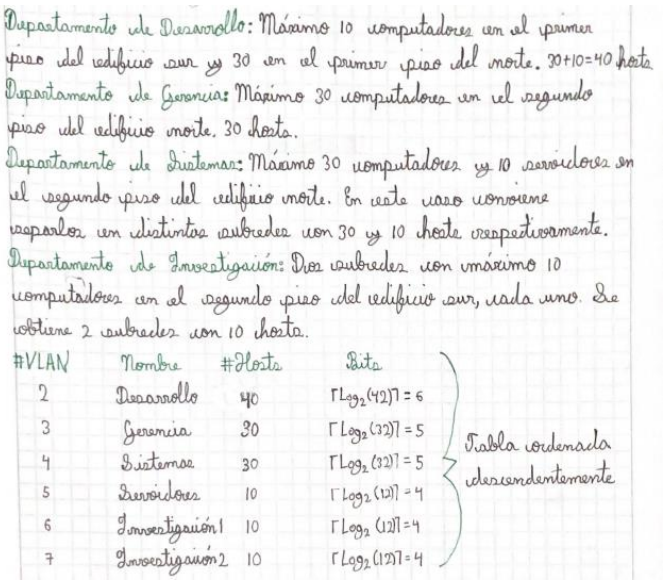


*Fig. 45. Cálculo de subredes y hosts, elaboración propia.*

Después, se verifica que la dirección original sea de red porque, de lo contrario, es imposible realizar subnetting; y se calcula la nueva máscara como: Nueva máscara = Anterior máscara + Bits prestados, y Bits prestados = Total - Anterior Máscara - Bits de subred.

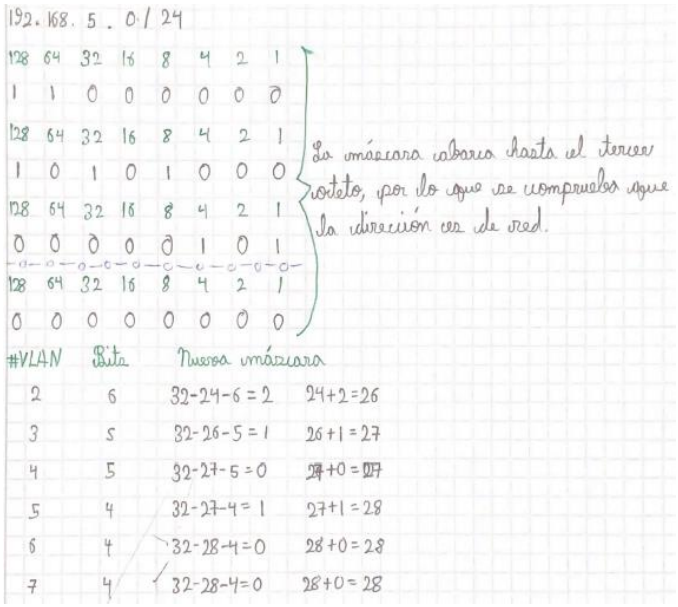


*Fig. 46. Cálculo de nuevas máscaras, elaboración propia.*

Para terminar el algoritmo, se transforma el último octeto no nulo de la máscara a binario y se calcula su valor para calcular Salto de red = Máximo valor de octeto - Último octeto y Dirección de red = Dirección Anterior + Salto Anterior. Nótese que la primera subred tiene la dirección de red original, pero con la nueva máscara y que no se calcula el salto de la última subred, ya que este valor no se sumará.

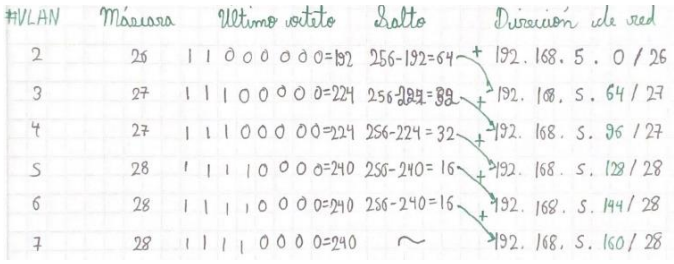


*Fig. 47. Cálculo de direcciones de red, elaboración propia.*

Una vez que se cuenta con las direcciones de red, sencillamente el Gateway será la siguiente a esta dirección y la dirección de broadcast será la anterior a la dirección de red de la siguiente subred o sumar la máxima cantidad de hosts a la dirección de Gateway. A continuación, se detalla la tabla completa de las direcciones a partir de VLSM.

**DIRECCIONAMIENTO**

| PUERTOS | VLAN | #VLAN | #HOST | BITS | IPv4 | MASCARA | GATEWAY IPv4 | IPv6 | GATEWAY IPv6 |
|---|---|---|---|---|---|---|---|---|---|
| 0-2 | DESARROLLO | 2 | 40 | 6 | 192.168.5.0/26 | 255.255.255.192 | 192.168.5.1 | FD00:3::/64 | FD00:3::1 |
| 3-5 | GERENCIA | 3 | 30 | 5 | 192.168.5.64/27 | 255.255.255.224 | 192.168.5.65 | FD00:4::/64 | FD00:4::1 |
| 6-8 | SISTEMASN | 4 | 30 | 5 | 192.168.5.96/27 | 255.255.255.224 | 192.168.5.97 | FD00:5::/64 | FD00:5::1 |
| 9-11 | SERVIDORES | 5 | 10 | 4 | 192.168.5.128/28 | 255.255.255.240 | 192.168.5.129 | FD00:6::/64 | FD00:6::1 |
| 1-7 | INVESTIGACION1 | 6 | 10 | 4 | 192.168.5.144/28 | 255.255.255.240 | 192.168.5.145 | FD00:1::/64 | FD00:1::1 |
| 1-7 | INVESTIGACION2 | 7 | 10 | 4 | 192.168.5.160/28 | 255.255.255.240 | 192.168.5.161 | FD00:2::/64 | FD00:2::1 |
| | INVESTIGACION | | | | 172.16.1.0/24 | 255.255.255.0 | 172.16.1.1 | FD00:7::/64 | FD00:7::1 |
| 12-15 | TRONCAL | 1 | | | | | | | |

*Tabla 1. Tabla de direcciones, elaboración propia.*

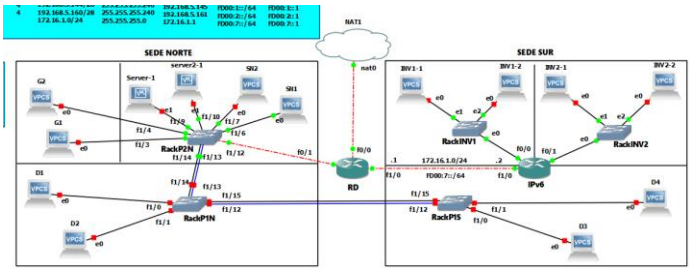


*Fig. 48. Topología propuesta, elaboración propia.*

En la Fig. 48 se puede observar la topología propuesta, donde se muestran 4 hosts de Investigación (INV1-1, INV1-2. INV2-1 y INV2-2), 4 hosts de Desarrollo (D1, D2, D3 y D4), 2 de Gerencia (G1 Y G2) y 2 de Sistemas (SN1 y SN2). Por otro lado, se visualizan dos switches pertenecientes a las VLANs de Investigación (RackINV1 y RackINV2). A su vez, tres switches de capa 3; uno para el piso 1 sur (RackP1S), otro para piso 1 norte (RackP1N) y finalmente, uno perteneciente al piso 2 norte (RackP2N). Además, se observan dos routers CISCO 3600 (RD y IPV6). Por último, se observa la NAT, donde su icono se encuentra entre las dos sedes, y dos servidores (Server-1 y server2-1) los cuales incluyen DHCP, DNS, WEB y LDAP mediante Virtual-Box.

En la implementación de IPv6 se aisló el segundo piso de la sede sur, donde se encuentran las dos VLANs de Investigación, de forma que haya comunicación con las demás VLANs sin dejar incomunicadas a los hosts de Investigación.

Con color rojo, se detallan los cables más importantes de la topología, estos corresponden al router RD, pues es el que se encarga de la mayor parte de la configuración.

Con color azul, se detallan los cables que corresponden a los enlaces redundantes.

**SPANNING-TREE**

| VLAN | #VLAN | PRIORIDAD |
|---|---|---|
| DESARROLLO | 2 | 300 |
| GERENCIA | 3 | 400 |
| SISTEMASN | 4 | 500 |
| SERVIDORES | 5 | 200 |
| TRONCAL | 1 | 100 |

*Fig. 49. Topología propuesta, elaboración propia.*

Siguiendo la tabla de la Figura 49, se implementó el protocolo de Spanning-Tree, donde se dejó como rack-raíz a RackP2N con las prioridades detalladas en la imagen. Así mismo, se dejó con prioridad secundaría al RackP1N, donde a la hora de implementar la configuración se le suma el valor de 100. Por último, al RackP1S se dejó como el más bajo de prioridad, sumándole el valor de 200.

En la figura 50 se puede ver la configuración por puerto NAT por DHCP para el router RD y en la figura 51 la propia configuración.

```
RD#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
RD(config)# interface fastEthernet 0/0
RD(config-if)#ip dhcp
% Incomplete command.

RD(config-if)#ip address dhcp
RD(config-if)#no shutdown
RD(config-if)#
*Mar  1 05:20:19.886: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 05:20:20.886: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
RD(config-if)#
*Mar  1 05:20:31.194: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned
 DHCP address 192.168.16.128, mask 255.255.255.0, hostname RD

RD(config-if)#$ DHCP address 192.168.16.128, mask 255.255.255.0, hostname RD
RD#
*Mar  1 05:20:48.202: %SYS-5-CONFIG_I: Configured from console by console
RD#{T
RD#E}M
RD#show ip interface brief
Interface                  IP-Address      OK? Method Status                Prot
ocol
FastEthernet0/0            192.168.16.128  YES DHCP   up                    up

FastEthernet0/1            unassigned      YES NVRAM  up                    up

FastEthernet0/1.2          192.168.5.1     YES NVRAM  up                    up

FastEthernet0/1.3          192.168.5.65    YES NVRAM  up                    up

FastEthernet0/1.4          192.168.5.97    YES NVRAM  up                    up

FastEthernet0/1.5          192.168.5.129   YES NVRAM  up                    up

FastEthernet1/0            172.16.1.1      YES NVRAM  up                    up
```

*Fig. 50. Configuración puerto NAT por DHCP, elaboración propia.*

```
RD#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
RD(config)#interface fastEthernet 0/0
RD(config-if)#ip nat source
                ^
% Invalid input detected at '^' marker.

RD(config-if)#ip nat outside
RD(config-if)#
*Mar  1 05:23:56.938: %LINEPROTO-5-UPDOWN: Line protocol on Interface NVI0, changed state to up
RD(config-if)#interface fastEthernet 0/1
RD(config-if)#ip nat inside
RD(config-if)#interface fastEthernet 1/0
RD(config-if)#ip nat inside
RD(config-if)#exit
RD(config)#access-list 1 permit 192.168.5.0 0.0.0.255
RD(config)#ip nat source list 1 interface fastEthernet 0/0 overload
RD(config)#exit
RD#write
*Mar  1 05:27:48.710: %SYS-5-CONFIG_I: Configured from console by console
RD#write
Building configuration...
[OK]
RD#
```

*Fig. 51. Configuración NAT, elaboración propia.*

## V. PROTOCOLOS DE PRUEBA E IMPLEMENTACIÓN

Como primera prueba para comprobar el funcionamiento de la arquitectura vamos a primero comprobar el funcionamiento del servidor DHCP abriendo la consola de dos máquinas SN1 y INV1-1 y posteriormente vamos a hacer un ping entre ambas para ver si existe comunicación, y como se ve en la figura 51 la prueba fue exitosa.

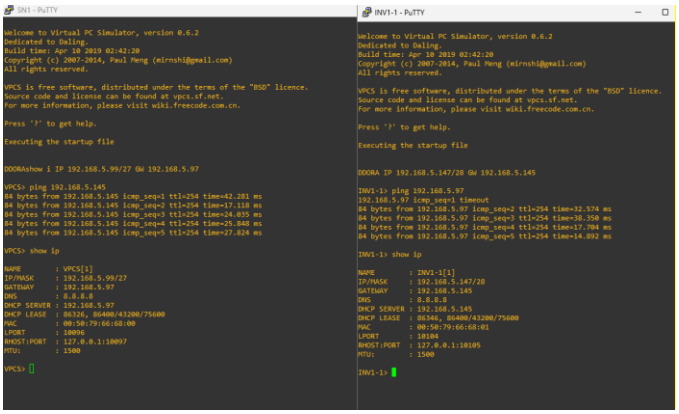


*Fig. 51. Ping IPv4 a IPv4 con DHCP, elaboración propia.*

Como segunda prueba para comprobar el funcionamiento de la arquitectura vamos a entrar en la consola de la máquina D3 y vamos a ver si se está ejecutando correctamente el protocolo ICMP, y como se ve en la figura 52 la prueba fue exitosa en lo que respecta al Request.

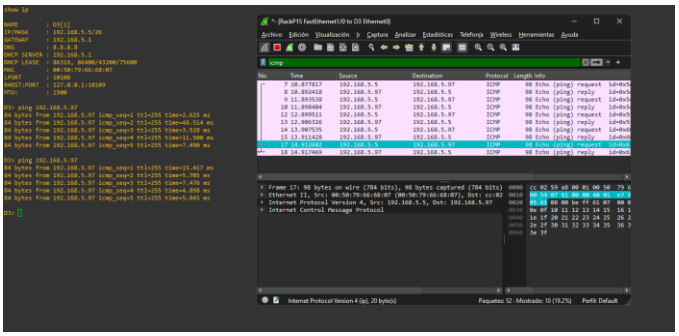

*Fig. 52. Ping y Request de IPv4 a IPv4 con WireShark, elaboración propia.*

Continuando la prueba anterior en la máquina D3, vamos a ver si se está ejecutando correctamente el protocolo ICMP, y como se ve en la figura 53 la prueba fue exitosa en lo que respecta al Reply, por lo que en conclusión si existe comunicación y con encapsulamiento de VLANs por el id que se muestra en ambas imágenes.

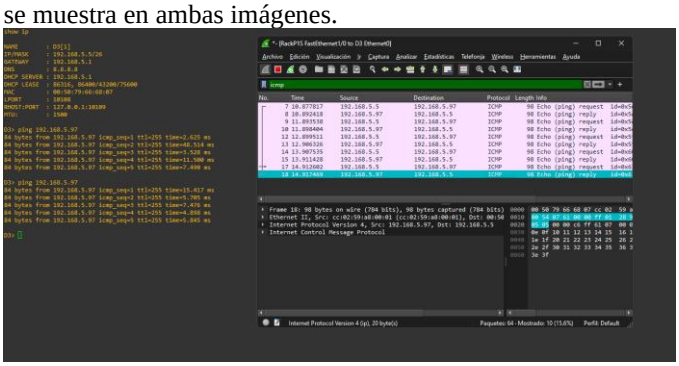

*Fig. 53. Ping y Reply de IPv4 a IPv4 con WireShark, elaboración propia.*

Y finalmente, para comprobar el buen funcionamiento de IPv6 no solo con las máquinas de investigación sino con cualquiera, procedemos a abrir la consola de las máquinas D3 y INV1-1 y después de asignarles la IPv6 correspondiente hacemos el ping y junto a WireShark podemos comprobar que si funciona.

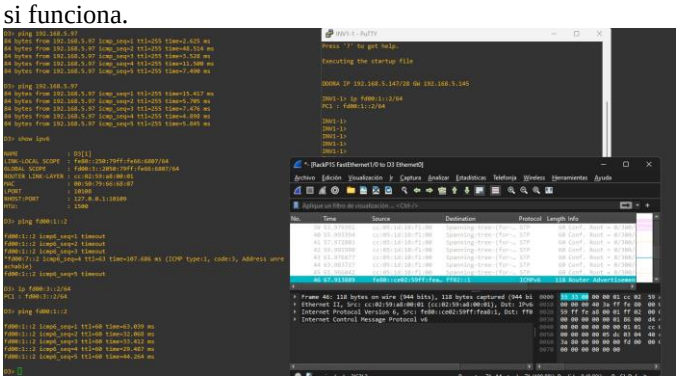

*Fig. 54. Ping IPv6 a IPv6 con WireShark, elaboración propia.*

### *VI. Video*

*El enlace para el video de Youtube es el siguiente: https://youtu.be/iIqIkYv8uPc*

## REFERENCIAS


[1] "¿Qué es un servidor web?" IONOS Digital Guide, https://www.ionos.es/digitalguide/servidores/know-how/servidor-web-definicion-historia-y-programas/ (accessed Apr. 10, 2024).

[2] Jesús, "Ventajas y desventajas de un servidor web," Tutoriales Dongee, https://www.dongee.com/tutoriales/ventajas-y-desventajas-de-un-servidor-web/ (accessed Apr. 10, 2024).

[3] A. S. Gillis, "What is LDAP (Lightweight Directory Access Protocol)?," Mobile Computing, https://www.techtarget.com/searchmobilecomputing/definition/LDAP#:~:text=LDAP%20(Lightweight%20Directory%20Access%20Protocol)%20is%20a%20software%20protocol%20for,internet%20or%20a%20corporate%20intranet. (accessed Apr. 10, 2024).

[4] Howes, T., Smith, M., & Good, G. S. (2003). Understanding and deploying LDAP directory services. Addison-Wesley Professional. (accessed Apr. 10, 2024).

[5] Montoya, C. E. G., Uribe, C. A. C., & Rodríguez, L. E. S. (2013). Seguridad en la configuración del servidor web Apache. Inge Cuc, 9(2), 31-38. (accessed Apr. 10, 2024).

[6] "What is DNS? – introduction to DNS," Amazon Web Service, https://aws.amazon.com/route53/what-is-dns/ (accessed Apr. 10, 2024).

[7] A. S. Gillis, "What is DHCP (dynamic host configuration protocol)?," Networking, https://www.techtarget.com/searchnetworking/definition/DHCP (accessed Apr. 10, 2024).

[8] RedesCiber - Tu Academia de Redes y Ciberseguridad (2022) *Diferencias entre hub, switch y router Explicado Rápido y fácil!, YouTube*. Available at: https://www.youtube.com/watch?v=Gky8-OVYmT4&t=327s (Accessed: *08 April 2024).*

[9] *A. S. Tanenbaum y D. J. Wetherall, *Redes de computadoras*, 5ta ed., Pearson Educación, 2011*

[10] *RedesCiber - Tu Academia de Redes y Ciberseguridad (2023) Diferencias entre un switch de capa 2, UN Switch de Capa 3 y un router, YouTube. Available at: https://www.youtube.com/watch?v=B10k4fa4hPU&t=6s (Accessed: 08 April 2024).*

[11] *Tintín, V. and Caiza, J. (2018) (PDF) arquitectura de redes de información. principios y Conceptos, Arquitectura de redes de información. Principios y conceptos. Available at: https://www.researchgate.net/publication/336003479_Arquitectura_de_redes_de_informacion_Principios_y_conceptos (Accessed: 13 April 2024).*

[12] *M., J.M. (2018) Diferencias Entre Redes Lan, man Y Wan, PC Solución. Available at: https://pc-solucion.es/tecnologia/diferencias-entre-redes-lan-man-y-wan/ (Accessed: 10 April 2024).*

[13] *Motiso, D. (2022) Network redundancy: Definition, types and how to improve it | indeed.com, indeed. Available at: https://www.indeed.com/career-advice/career-development/network-redundancy (Accessed: 08 April 2024).*

[14] *Sheldon, R. (2021) What is Spanning Tree Protocol?, Networking. Available at: https://www.techtarget.com/searchnetworking/definition/spanning-tree-protocol (Accessed: 09 April 2024).*

[15] *Understand and configure STP on Catalyst Switches (2023) Cisco. Available at: https://www.cisco.com/c/en/us/support/docs/lan-switching/spanning-tree-protocol/5234-5.html (Accessed: 08 April 2024).*

[16] *Encabezado de Paquetes IPv4 Cisco - Eclassvirtual - cursos cisco en línea (2022) eClassVirtual. Available at: https://eclassvirtual.com/encabezado-de-paquetes-ipv4-cisco/ (Accessed: 10 April 2024).*

[17] *W0lf_F4ng (2020) Fundamentos de IPv6 – Parte 1: W0lf_f4ng, Fundamentos de IPv6 – Parte 1 |. Available at: https://www.w0lff4ng.org/fundamentos-de-ipv6-parte-1/ (Accessed: 10 April 2024).*